

Create techy_vault table

```
CREATE DATABASE IF NOT EXISTS techy_vault;
USE techy_vault;
```

Create account_holder table

```
CREATE TABLE account_holder (
    account_holder_id INT AUTO_INCREMENT PRIMARY KEY,
    first_name VARCHAR(50) NOT NULL,
    last_name VARCHAR(50) NOT NULL,
    middle_name VARCHAR(50),
    account_id INT,
    phone_number VARCHAR(20),
    email VARCHAR(100),
    password VARCHAR(255) NOT NULL,
    created_at DATETIME DEFAULT CURRENT_TIMESTAMP,
    updated_at DATETIME DEFAULT CURRENT_TIMESTAMP ON UPDATE
CURRENT_TIMESTAMP
);
```

Create bank_teller table

```
CREATE TABLE bank_teller (
    teller_id INT AUTO_INCREMENT PRIMARY KEY,
    first_name VARCHAR(50) NOT NULL,
    last_name VARCHAR(50) NOT NULL,
    middle_name VARCHAR(50),
    account_id INT,
    phone_number VARCHAR(20),
    email VARCHAR(100),
    password VARCHAR(255) NOT NULL,
    created_at DATETIME DEFAULT CURRENT_TIMESTAMP,
    updated_at DATETIME DEFAULT CURRENT_TIMESTAMP ON UPDATE
CURRENT_TIMESTAMP
);
```

Create bank_account table

```
CREATE TABLE bank_account (
    account_id INT AUTO_INCREMENT PRIMARY KEY,
    name VARCHAR(100) NOT NULL,
    account_number VARCHAR(20) NOT NULL UNIQUE,
    account_balance DECIMAL(15, 2) DEFAULT 0.00,
    is_active BOOLEAN DEFAULT TRUE,
    created_at DATETIME DEFAULT CURRENT_TIMESTAMP,
    updated_at DATETIME DEFAULT CURRENT_TIMESTAMP ON UPDATE
CURRENT_TIMESTAMP
);
```

Create fund_transfer_internal table

```
CREATE TABLE fund_transfer_internal (
    fund_transfer_internal_id INT AUTO_INCREMENT PRIMARY KEY,
    transaction_no VARCHAR(50) NOT NULL UNIQUE,
    recipient_account_id INT NOT NULL,
    transaction_amount DECIMAL(15, 2) NOT NULL,
    transaction_date DATETIME DEFAULT CURRENT_TIMESTAMP,
    FOREIGN KEY (recipient_account_id) REFERENCES bank_account(account_id)
);
```

Create fund_transfer_external table

```
CREATE TABLE fund_transfer_external (
    fund_transfer_external_id INT AUTO_INCREMENT PRIMARY KEY,
    sender_account_id INT NOT NULL,
    recipient_account_id VARCHAR(50) NOT NULL,
    recipient_bank_name VARCHAR(100) NOT NULL,
    transaction_no VARCHAR(50) NOT NULL UNIQUE,
    transaction_amount DECIMAL(15, 2) NOT NULL,
    transaction_date DATETIME DEFAULT CURRENT_TIMESTAMP,
    FOREIGN KEY (sender_account_id) REFERENCES bank_account(account_id)
);
```

Create receive_external_transfer table

```
CREATE TABLE receive_external_transfer (
    receive_external_transfer_id INT AUTO_INCREMENT PRIMARY KEY,
    sender_account_id VARCHAR(50) NOT NULL,
    recipient_bank_name VARCHAR(100) NOT NULL,
    recipient_account_id INT NOT NULL,
    transaction_no VARCHAR(50) NOT NULL UNIQUE,
    transaction_amount DECIMAL(15, 2) NOT NULL,
    transaction_date DATETIME DEFAULT CURRENT_TIMESTAMP,
    FOREIGN KEY (recipient_account_id) REFERENCES bank_account(account_id)
);
```

Create teller_transaction table

```
CREATE TABLE teller_transaction (
    teller_transaction_id INT AUTO_INCREMENT PRIMARY KEY,
    transaction_type VARCHAR(50) NOT NULL,
    teller_id INT NOT NULL,
    account_id INT NOT NULL,
    transaction_amount DECIMAL(15, 2) NOT NULL,
    transaction_date DATETIME DEFAULT CURRENT_TIMESTAMP,
    FOREIGN KEY (teller_id) REFERENCES bank_teller(teller_id),
    FOREIGN KEY (account_id) REFERENCES bank_account(account_id)
);
```

Create fund_deposit table

```
CREATE TABLE fund_deposit (
```

```

    fund_deposit_id INT AUTO_INCREMENT PRIMARY KEY,
    teller_transaction_id INT NOT NULL,
    account_id INT NOT NULL,
    transaction_amount DECIMAL(15, 2) NOT NULL,
    deposit_date DATETIME DEFAULT CURRENT_TIMESTAMP,
    FOREIGN KEY (teller_transaction_id) REFERENCES
teller_transaction(teller_transaction_id),
    FOREIGN KEY (account_id) REFERENCES bank_account(account_id)
);

```

Create fund_withdraw table

```

CREATE TABLE fund_withdraw (
    fund_withdraw_id INT AUTO_INCREMENT PRIMARY KEY,
    teller_transaction_id INT NOT NULL,
    account_id INT NOT NULL,
    transaction_amount DECIMAL(15, 2) NOT NULL,
    withdraw_date DATETIME DEFAULT CURRENT_TIMESTAMP,
    FOREIGN KEY (teller_transaction_id) REFERENCES
teller_transaction(teller_transaction_id),
    FOREIGN KEY (account_id) REFERENCES bank_account(account_id)
);

```

Create teller_close_account table

```

CREATE TABLE teller_close_account (
    teller_close_account_id INT AUTO_INCREMENT PRIMARY KEY,
    teller_id INT NOT NULL,
    account_id INT NOT NULL,
    reason VARCHAR(255) NOT NULL,
    withdraw_date DATETIME DEFAULT CURRENT_TIMESTAMP,
    FOREIGN KEY (teller_id) REFERENCES bank_teller(teller_id),
    FOREIGN KEY (account_id) REFERENCES bank_account(account_id)
);

```

Update account_holder to add foreign key after all tables are created

```

ALTER TABLE account_holder
ADD CONSTRAINT fk_account_holder_account
FOREIGN KEY (account_id) REFERENCES bank_account(account_id);

```

Update bank_teller to add foreign key after all tables are created

```

ALTER TABLE bank_teller
ADD CONSTRAINT fk_teller_account
FOREIGN KEY (account_id) REFERENCES bank_account(account_id);

```